REMARKS

No claims are amended. Claims 32-34 are added. Claims 1-34 are pending. In view of the following remarks, Applicant respectfully requests reconsideration and allowance of the subject application.

The § 112 Rejections

Claims 1-31 stand rejected under 35 U.S.C. § 112, first paragraph, as containing subject matter that is, in the Office's opinion, not enabled. Specifically, the Office argues that "determining whether an attack pattern is a disclosure attack, integrity attack, and/or a denial of service attack" is not enabled. Applicant respectfully disagrees and traverses the Office's rejections.

Applicant respectfully draws the Office's attention to the specification, page 1, line 18, through page 3, line 6, reproduced below:

> In the past, malicious individuals have used input strings that are intended for use by Web servers to attack the servers. These individuals will typically try to find an input string that causes the Web server or, perhaps its operating system, to perform in a manner that is inconsistent with simply processing legitimate client requests and returning authorized resources to the client. ***Input strings that have been used in the past to attack Web servers seem to come in an ever-changing number of varieties and formats***. The various attacks that can be waged against a Web server can be categorized as disclosure attacks, integrity attacks, and denial of service attacks.
>
> A ***disclosure attack*** takes place when an individual attacks a web site and attempts to read information that they are not authorized to read. For example, there may be some executable code at the server that an individual is not authorized to view. Yet, by providing an input string that causes the server to malfunction, the individual actually gets to view the executable code. Consider, for example, Active Server Pages. Active Server Pages can allow Web developers to use scripting languages like Visual Basic Script and JScript to pass information to various components that contain logic for accessing databases, instruct the components to perform a

programmed action, and return the results of the programmed action. The individual is only authorized, and supposed to view the results of the programmed action. Yet, by using particular inappropriate input strings it may be possible for the individual to view the code that produces the results.

An ***integrity attack*** is similar to a disclosure attack in that an individual can gain access to unauthorized information. In addition to gaining access to the information, however, integrity attacks involve the manipulation of data or information that is being viewed. This is particularly problematic because the changed, now-invalid information can potentially further compromise an already-compromised Web server.

A ***denial of service attack*** is an attack that can cause a decrease in the quality of service or, ultimately, can cause the server to crash. This can adversely impact the server's ability to service other legitimate clients thereby leading to undesirable downtime and customer dissatisfaction.

Many of these types of attacks can be traced directly to the mishandling of an input string that was provided to the Web server. A need exists to deal with problematic input strings in a flexible, quick and convenient manner.

As noted above, Applicant describes different examples of attacks that can be waged on a server. Applicant also instructs that input strings that have been used in the past to attack Web servers seem to come in an ever-changing number of varieties and formats. Further on in the Specification, particularly starting on page 9 at line 10, Applicant describes an example of a problematic input string and the effects that such an input string might have. See, e.g. page 9, lines 21-25 through page 10, line 5, reproduced below for the convenience of the Office:

> **Input String Screening**
>
> Aspects of the invention enable an input string that is provided by a client to be screened before it is processed by the Web server. An "input string" is a URL or other string that is intended for use by the Web server. Screening the input strings ensures that problematic input strings are identified and handled appropriately so that the risk of adversely impacting

the Web server is reduced. As an example of a problematic input string consider the following URL input string:

http://www.foo.com/../../../../boot.ini

Assume that data that is associated with www.foo.com is stored in a directory "c:\wwroot\stuff\data". The ".." that appears in the URL input string after the www.foo.com specification can cause the server to move up in the hierarchical directory from "c:\wwroot\stuff\data" by one directory. A series of ".." in the URL input string can cause the server to move up in the hierarchical directory a number of times until it reaches the root directory, in this case the "c:" directory. At this point it might be possible to get access any files in the root directory such as the specified "boot.ini" file. This file might constitute a file that describes how the computer is designed to boot. In this case, a user would be able to view and possibly manipulate an unauthorized file.

Applicant provides an additional example of a problematic input string starting on page 10, line 4, which is reproduced below for the convenience of the Office:

As another example, consider the following URL input string:

http://www.foo.com/datalookup.asp::$DATA

In this example, it is possible that the server might not understand the "::$DATA" portion of this input string, but that the string portion has a special meaning to the operating system on which the server is executing. As a consequence, the operating system might cause unauthorized files to be accessible to the user.

The Specification then goes on to instruct why these types of strings are problematic and additional characteristics of input strings that can be problematic. Specifically, consider page 10, line 15 through page 11, line 3, reproduced in its entirety below:

In both of these examples, the input string can be characterized as containing a pattern that is problematic to the Web server. It is problematic because it can cause the Web server or its operating system to behave in a manner that is inconsistent with returning only authorized resources to a client. In this document, such patterns are referred to as "attack patterns" because they effectively enable an attack on the server. In the above two examples, the attack patterns are constituted by the ".." and "::$" portions of the input string.

In addition to these exemplary attack patterns, there are also input string characteristics that can be indicative of an attack pattern. One such characteristic is if the input string does not contain an alphabetical character at its end. Another characteristic is whether the input string contains any specific "operators" that are inappropriate for an input string. Examples include the operators "|", "<", ">", and "&". Any input string that is found to satisfy the characteristics that are indicative of an attack pattern are likely to be problematic for the server.

Having explored different types of attacks that can be waged on a server and given specific examples of problematic input strings and associated characteristics, the Specification then provides a description of a set of tools that can be used to address these types of problematic input strings starting on page 11 at line 5, aspects of which are reproduced below:

Web Server Pattern Matching

Fig. 3 shows a flow diagram that describes steps in an input string screening method for a Web server in accordance with one embodiment of the invention. ***Step 200 determines an attack pattern that can be used to attack a Web server. One way in which this determination can be made is by simply observing over time, which attacks on a Web server are successful. Another way to determine an attack pattern is to recognize that there are input string characteristics that can be problematic for a Web server. For example, input strings that contain the pattern ".." can be problematic because they might enable an individual to inappropriately "walk" up a directory tree. Additionally, attack patterns can be determined by recognizing that there are certain characters that***

are simply not appropriate for inclusion in an input string. Examples of certain operators were given above.

With one or more attack patterns having been determined, step 202 defines a search pattern that can be used to detect the attack pattern. A search pattern is an expression that is compared with input strings to determine whether there is a matching search pattern in the input string. In the described embodiment, a search pattern can be formatted syntactically in a manner that allows specification of both identity and variability among constituent parts of an input string. Thus, the search pattern can include literal parts that call for an exact character-by-character match between those parts and corresponding parts of the input string, and variable parts that allow for inexact matches or no match at all between those parts and corresponding parts of the input string. An input string is said to "match" a search pattern if the search pattern is found anywhere within the input string as specified by the search pattern. In the described embodiment, one or more search patterns are specified as regular expressions. In a regular expression, each character matches itself, unless it is one of a number of special characters that indicate variable characters in the input string. An example subset of regular expression definitions and their meanings is given below:

Pattern	Meaning
.	Matches an arbitrary character
(...)	Groups a series of pattern elements to a single element
^	Matches the beginning of the target
+	Matches the preceding pattern elements one or more times. For example, ba+c matches bac, baac, but not bc.
$	Matches the end of the line. For example, 100$ matches 100 at the end of a line.
[...]	Denotes a class of characters to match; [^...] negates the class. For example, b[aeiou]d matches bad, bed, bid, bod, and bud (but not bead or beed); and r[eo]+d matches red, rod, reed, rood, reod, roed, reood, rocod, etc.
[^]	Matches any character except those following the caret (^) character in the brackets, or any of an ASCII range of characters separated by a hyphen (-). For example, x[^0-9] matches xa, xb, xc, and so on, but not x0, x1, x2, and so on.
(...|...)	Matches one of the alternatives
?	Matches the preceding character zero or one time.
*	Matches the preceding character zero or more times. For example, ba*c matches bc, bac, baac, and so on.
{}	Matches any sequence of characters between the escaped braces. For example, {ju}+fruit matches jufruit, jujufruit, but not ufruit, jfruit, or ujfruit.
\	Removes the pattern match characteristics from the special characters listed above. For example, 100$ matches 100 at the end of a line, but 100\$ matches the character string 100$ anywhere on a line.

By defining search patterns as described above, flexibility and extensibility are enhanced by enabling a system administrator to define a search pattern in terms of a generalized regular pattern that reflects an attack pattern of which the system administrator has recently become aware. The definition of search patterns in this manner is timely because the search patterns can be defined almost as soon as the attack patterns are detected, without the need to hardcode specific patterns.

In the described embodiment, patterns can be collected into collections of patterns as more and more patterns are observed or determined. Accordingly, step 204 adds the pattern defined in step 202 to such a collection. The collection of patterns can be stored and maintained in memory. In the described embodiment, the collection is adapted for addition to, delction of, or modification of the patterns that it contains. This facilitates the overall extensibility of the collection of patterns. In the described embodiment, steps 200-204 can be implemented using an administrative tool or some other suitable interface.

Step 206 receives an input string from the client that is intended for use by the Web server, and step 208 evaluates the input string using one or more of the search patterns. Step 210 determines whether any of the attack patterns are present in the input string. An attack pattern is present if a match is found for the search pattern in the input string. If there are no attack patterns present in the input string, then step 212 processes the input string or request that is associated with the input string. Where an input string comprises a URL, processing can include retrieving an appropriate resource, i.e. a Web page, and returning it to the client. If, on the other hand, there is an attack pattern that is identified to be associated with the input string (i.e. an attack pattern is found in the input string that matches the search pattern), then step 214 implements a remedial action. Remedial actions can be any actions that are associated with minimizing or eliminating the effect that an attack pattern can have on the Web server. In but one example, this can include denying a request that is associated with the input string. For example, in the case of an input string that is a URL, this could mean returning an error message to the client to the effect that the request could not be executed.

Thus, this portion of the Specification, building on the specific examples and discussion provided above, then instructs how one might go about determining an attack pattern. For example, one way is to simply observe, over time, which

attacks are successful. See, e.g. page 11, lines 9-10. Another way, is to recognize that there are input string characteristics that are problematic for a Web server. See, e.g. page 11, lines 10-12. Yet another way to determine an attack pattern is to recognize that there are certain characters that are simply not appropriate for inclusion in an input string. See, e.g. page 11, 14-17.

The Specification then goes on to note that the disclosed search pattern definition tools can enable a system administrator to define a search pattern in terms of a generalized regular pattern that reflects an attack pattern of which the system administrator has recently become aware. See, e.g. page 12, lines 21-24.

Effectively, the Specification describes different types of attacks and provides a set of tools that allows an administrator to flexibly design a search pattern responsive to observing a specific attack. These different types of attacks are well within the understanding of a person of skill in the art. Additionally, recognizing problematic input strings that are associated with a particular type of attack is also well within the understanding of a person of skill. ***This is particularly the case after an actual attack when the system administrator would have access to the actual input string that caused the attack.*** Given this, designing a search pattern to search for the identified problematic input string is additionally within the grasp of a person of skill based on the teachings of the Specification.

It is not Applicant's intent, nor is it practically feasible to describe each and every problematic input string that might exist and be used to attack a server. Rather, one goal of the various embodiments is to provide a set of tools which, once a problematic input string has been identified, can be used to address and mitigate the effects of the input string.

As such, Applicant respectfully submits that this disclosure is enabling for all of the attacks described in the Specification.

Claims 1-17 and 22-31 stand rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for "failing to particularly point out and distinctly claim the subject matter which Applicant regards as the invention." In making out this rejection, the Office argues that the phrase ". . . content that is designed to constitute . . ." renders the claims indefinite because it makes it unclear as to whether the content must actually be one of the enumerated types of attack patterns. Applicant respectfully disagrees and traverses the Office's rejections.

The claim language at issue is: ". . . the attack pattern comprising content that is designed to constitute ***one or more of*** a disclosure attack, an integrity attack or a denial of service attack on the Web server." Applicant respectfully submits that the claim language is clear and is in fact a valid Markush group. Accordingly, Applicant respectfully requests the Office to withdraw these rejections.

The § 103 Rejections

Claims 1-11 and 13-30 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,884,033 to Duvall et al (hereinafter, "Duvall") in view of U.S. Patent No. 6,421,781 to Fox et al (hereinafter, "Fox").

Claims 12 and 31 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Duvall in view of Fox and Oliver et al., "Building a Windows NT 4 Internet Server", 1996, p. 203.

The § 103 Standard

To establish a prima facie case of obviousness, three basic criteria *must* be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992); *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Second, there must be a reasonable expectation of success. *In re Merck & Co., Inc.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, not in applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1439 (Fed. Cir. 1991).

Hence, when patentability turns on the question of obviousness, the search for and analysis of the prior art includes evidence relevant to the finding of whether there is a teaching, motivation, or suggestion to select and combine the references relied on as evidence of obviousness. See, e.g., *McGinley v. Franklin Sports, Inc.*, 262 F.3d 1339, 1351-52, 60 USPQ2d 1001, 1008 (Fed. Cir. 2001) ("the central question is whether there is reason to combine [the] references," a question of fact drawing on the Graham factors). The mere fact that references *can* be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990).

"The factual inquiry whether to combine references must be thorough and searching." *Id.* It must be based on objective evidence of record. This precedent has been reinforced in myriad decisions, and cannot be dispensed with. See, e.g., *Brown & Williamson Tobacco Corp. v. Philip Morris Inc.*, 229 F.3d 1120, 1124-25, 56 USPQ2d 1456, 1459 (Fed. Cir. 2000) ("a showing of a suggestion, teaching, or motivation to combine the prior art references is an 'essential component of an obviousness holding'") (quoting *C.R. Bard, Inc., v. M3 Systems, Inc.*, 157 F.3d 1340, 1352, 48 USPQ2d 1225, 1232 (Fed. Cir. 1998)); *In re Dembiczak*, 175 F.3d 994, 999, 50 USPQ2d 1614, 1617 (Fed. Cir. 1999) ("Our case law makes clear that the best defense against the subtle but powerful attraction of a hindsight-based obviousness analysis is rigorous application of the requirement for a showing of the teaching or motivation to combine prior art references."); *In re Dance*, 160 F.3d 1339, 1343, 48 USPQ2d 1635, 1637 (Fed. Cir. 1998) (there must be some motivation, suggestion, or teaching of the desirability of making the specific combination that was made by the applicant); *In re Fine*, 837 F.2d 1071, 1075, 5 USPQ2d 1596, 1600 (Fed. Cir. 1988) ("'teachings of references can be combined only if there is some suggestion or incentive to do so.'") (emphasis in original) (quoting *ACS Hosp. Sys., Inc. v. Montefiore Hosp.*, 732 F.2d 1572, 1577, 221 USPQ 929, 933 (Fed. Cir. 1984)); *In re Fritch*, 23 USPQ2d 1780, 1784 (Fed. Cir. 1992) ("It is impermissible to use the claimed invention as an instruction manual or 'template' to piece together the teachings of the prior art so that the claimed invention is rendered obvious. [O]ne cannot use hindsight reconstruction to pick and choose among isolated disclosures in the prior art to deprecate the claimed invention.") (quoting *In Re Fine*, 837 F.2d 1071, 1075, 5 USPQ2d 1596, 1600 (Fed. Cir. 1988)).

The need for specificity pervades this authority. See, e.g., *In re Kotzab*, 217 F.3d 1365, 1371, 55 USPQ2d 1313, 1317 (Fed. Cir. 2000) ("particular findings must be made as to the reason the skilled artisan, with no knowledge of the claimed invention, would have selected these components for combination in the manner claimed").

A factor cutting against a finding of motivation to combine or modify the prior art is when the prior art *teaches away* from the claimed combination. A reference is said to teach away when a person of ordinary skill, upon reading the reference, would be led in a direction divergent from the path that the applicant took. *In re Gurley*, 31 USPQ 2d 1130, 1131 (Fed. Cir 1994).

In addition, the references must either be in the field of the inventor's endeavor, or reasonably pertinent to the specific problem with which the inventor was involved. *In re Deminski*, 230 USPQ 313, 315 (Fed. Cir. 1986). Put another way, the references must be in an art *analogous* to that of the invention.

Applicant disagrees with the Office's obviousness rejections and respectfully submits that the Office has not made out a *prima facie* case of obviousness. Accordingly, Applicant respectfully requests withdrawal of these rejections.

The Duvall Reference

The reference to **Duvall** discloses a ***client***-based filtering system. The system allows a user to filter material received over the Internet that is ***personally objectionable***, whether that material is sexually explicit, violent, politically extreme, or otherwise, depending on the user's ***individual tastes and sensitivities***.

The filter compares portions of incoming and/or outgoing messages to filtering information in a filter database and determines whether to block or allow incoming and/or outgoing transmissions of messages in response to the comparison. In response to a match between the portion of the message and the filtering information, the system can employ one of a number of different specified blocking options.

The Fox Reference

Fox discloses what it considers a "secure" push server. The push server is used for sending notifications to wireless clients. An information service provider initiates a request to the push server that includes updated information and a site certificate. The push server examines the site certificate to determine the identity of the requester. If any URLs are referred to in a notification request, the push server ensures that the URL refers only to information located within the ***specific domain name*** identified in the certificate or an immediate superdomain of the specific domain name identified in the certificate. For example, if a site certificate identifies the domain name as push.www.unwiredplanet.com, the accompanying notification may ***only*** contain the ***exact same*** domain name or www.unwiredplanet.com (the immediate superdomain of push.www.unwiredplanet.com). Referring to Fig. 5, Fox explains that if the domain name of the URL contained in the notification does not ***exactly match*** the domain name identified in the certificate or its immediate superdomain (step 580), then the request is denied at step 590.

As such, Fox is merely performing a ***literal string comparison*** between the domain name of the URL contained in the notification and the domain name specified in the certificate (or its immediate superdomain).

Claims 1-6

Claim 1 recites a Web server input string screening method comprising [emphasis added]:

- determining an attack pattern that can be used to attack a Web server, the attack pattern comprising content that is designed to constitute one or more of a disclosure attack, an integrity attack or a denial of service attack on the Web server;
- defining a search pattern that can be used to detect the attack pattern, the search pattern being defined in a manner that permits ***variability among its constituent parts***;
- receiving an input string that is intended for use by a Web server;
- evaluating the input string using the search pattern to ascertain whether the attack pattern is present; and
- implementing a remedial action if an attack pattern is found that matches the search pattern.

In making out the rejection of this claim, the Office states that Duvall only discloses filtering of URL's that are related to material that is objectionable, depending upon the user's tastes and sensitivities. Applicant agrees. The Office further states that Duvall does not disclose the filtering of attacks on a system, such as a disclosure attack, integrity attack, or a denial of service attack. Again, Applicant agrees.

The Office then argues that Fox "discloses the parsing and checking of an incoming URL against a list of acceptable domains and variations thereof, and

notes that this protects against denial-of-service attacks." The Office cites to column 11, line 15, to column 14, line 4, for support, which is reproduced below:

> The present invention also examines the content of new notifications. Specifically, the push server examines notifications to see if any Uniform Resource Locators (URLs) are referenced in new notification requests. If any URLs are referred to, those URLs should be closely associated with the domain name of the entity that sent the notification request. The reason for this test is that an authorized authenticated entity should not be able to refer to information outside of its control.
>
> For example, one type of notification that may be sent is an "alert" that notifies the user of an important event. An alert consists of a brief text title, a URL, and a token that indicates how the user should be notified (i.e. a beep, flash, vibration, etc.). Upon receiving an alert, the client software in the wireless device places the text title into a status page dedicated to alerts. The client software also links the text title to the URL that was provided. The user may subsequently select the title text and therefore request the content associated with the linked URL. A malicious entity could abuse this feature by sending an alert with a "new email" text title and providing a URL that points to a list of forged email messages. The user would thus be tricked into viewing a set of false email messages.
>
> An attacker could also abuse the notification feature by sending a flood of notification requests that refer to a URL associated with a third party's server that the attacker wishes to attack. This flood of notifications would cause the push server to repetitively access the specified URL thereby degrading the performance of the server associated with the URL. Therefore, the flood of notifications would constitute a denial of service attack that would degrade the operation of the third party's site.
>
> An attacker could also abuse the notification feature by sending bogus cache invalidation requests. Each wireless client device has a cache that stores information that the wireless client device has received. In one embodiment, each piece of stored information may be associated with a URL where the piece of information originated. An attacker could send notification requests that perform cache invalidation on a URL outside of the domain of the attacker. This

cache invalidation request would invalidate valid information stored in the wireless client device. Such an attack would degrade the performance of the wireless client device (by invalidating valid information), the push server (by having to process the bogus notification), and the server associated with the URL (since an unnecessary cache update would be performed).

To prevent such abuses, the present invention only allows a notification to reference servers closely associated with the domain name listed in the certificate that accompanied the notification requested. One embodiment of the present invention requires new notifications to refer only to information located within the specific domain name identified in the certificate that accompanied the request or an immediate superdomain of the specific domain name identified in the certificate that accompanied the request. For example, if a a new notification request is accompanied by a site certificate that identifies the internet domain name "push.www.unwiredplanet.com" as the sender, then the following URLs may be placed in the notification:

http://push.www.unwiredplanet.com/info.txt (the same domain name)

https://www.unwiredplanet.com/abc (the superdomain)

However, the following URLs would not be acceptable:

http://home.www.unwiredplanet.com/push.txt (different domain)

https://unwiredplanet.com push.html (not the immediate superdomain)

This requirement will prevent an authorized authenticated entity from sending information located in a site outside of their control.

In one embodiment of the present invention, there are two different types of notifications: Pull notifications and Push notifications. Pull notifications refer to updated information that exists at a location that is specified using a URL. The URL is specified in a header field of the request. Push notifications contain a information payload that specifies updated information. However, the information payload of a push notification may include a URL that refers to outside information. Thus, both push and pull notifications must be checked.

To verify the content of notifications in an embodiment that uses both push and pull notifications, the present invention puts limitations on the URLs that may be used in the add notification request. Specifically, all URLs in a header field must be absolute and complete through the net_loc portion such that a domain name can be extracted from the URL and compared with a domain name from the site certificate. The net_loc portion, as defined in the Internet Engineering Task Force's (IETF) Request For Comments (RFC) document number 1808, is the domain name address portion of an internet server. For example, in the following Uniform Resource Locator (URL):

http://www.unwiredplanet.com/index.html

The www.unwiredplanet.com section of the Uniform Resource Locator (URL) is the net_loc portion of the URL. Furthermore, any URLs in the body of a push notification should be relative URLs such that those relative URLs are combined with the absolute URL in the header which was tested as set forth above.

Content Verification Embodiment

FIG. 5 illustrates a flow diagram of one possible embodiment of a push server system that ensures that the content of new notifications and maintenance requests are legitimate. It should be noted that the embodiment of FIG. 5 represents only one possible method of implementing the teachings of the present invention. For example, the steps listed in FIG. 5 may be performed in different order than presented in FIG. 5.

Referring to step 510 of FIG. 5, an authorized authenticated request has been received at a push server. The contents of the authorized authenticated request are examined to see if the request is a maintenance request that may refer to one or more earlier notifications or if the request is an add notification request that may refer to a URL that needs to be tested.

If, at step 520, the push server determines that the request is a maintenance request that may refer to one or more earlier notifications, then the push server proceeds to step 530. At step 530, the push server attempts to locate any previous notifications that the maintenance request concerns. Detailed information on how the push

server locates earlier notifications can be found in the parent U.S. patent application entitled "Method and Apparatus for Informing Wireless Clients about Updated Information" having Ser. No. 09/071,377 filed on Apr. 30, 1998 which is hereby incorporated by reference. If no matching notification is found, then the push server informs the requestor that no matching notification was found.

Assuming that at least one matching notification was found, then the push server, at step 560, compares the domain name associated with the matching notification with the domain name from the site certificate accompanying the maintenance request. Note that the domain name from the site certificate that accompanied the add notification request that created the matching notification was stored along with the notification. If the two domain names match exactly, then the maintenance request will be processed at step 600. Otherwise, if the domain names do not match, then the maintenance request is denied at step 610.

Referring back to step 520, if the request is a new add notification request then the push server proceeds to step 540. Each new add notification request must be examined to be sure that the notification does not refer to information outside of the sender's control. In the particular embodiment of FIG. 5, the push server ensures that all Uniform Resource Locators (URLs) in a notification are closely associated with the domain name of the entity that sent the notification request. In one embodiment that will be described, absolute URLs in header fields are tested and any URLs within a body of a notification request must only contain relative URLs that will be completed using an absolute URL in the header.

At step 540, the push server determines if there are any Uniform Resource Locators (URLs) in the header of the new notification request. If there are no URLs in the new notification request, then the push server proceeds to step 600 and processes the new notification request.

If there is a URL in the new notification request, then that URL needs to be checked. Step 550 tests to see if an absolute URL is provided. If the URL is not absolute, then the request is denied at step 590.

After determining that the Uniform Resource Locator (URLs) is absolute, step 570 tests to see if the URL is complete through the

> net_loc portion of a URL. If the enclosed URL does not include a non-empty and well-formed net_loc portion, then the request is denied at step 590. The request is denied since without a net_loc, the push server will not be able to verify that the URL is closely associated with the domain name that has already been authenticated.
>
> Finally, if the URL in the new notification is absolute and includes a net_loc, then the push server compares the net_loc with the domain name that was obtained from the site certificate that accompanied the new add notification request. The net_loc must be closely associated with the authenticated domain name from the site certificate. In one embodiment, the Internet address must match the immediate domain name identified in the site certificate or the immediate superdomain of the domain name identified in the certificate.
>
> Step 580 performs the step of comparing the net loc portion of the URL. If the net_loc does not exactly match the domain name identified in the certificate or the superdomain of the domain name identified in the certificate, then the request is denied at step 590. Note that the comparison is case insensitive. If the net_loc matches either the domain name identified in the site certificate or the superdomain of the domain name identified in the site certificate, then the request is processed at step 600.

First, Applicant respectfully submits that there would have been no motivation to combine the Duvall and Fox disclosures. The Office argues that it would have been obvious "to use the invention of Duvall by checking a URL against domain names, as disclosed by Fox, in order to protect against abusive denial-of-service attacks." As noted above, Duvall filters ***subjectively objectionable material*** on the ***client*** side. Because Duvall deals only with the client side, Duvall would have no reason to guard against ***denial of service attacks*** on the ***server*** side. Consequently, there would be no reason for a person skilled in the art to look to Fox's disclosure. In fact, it is unclear to Applicant how Fox's

disclosure could *possibly* be incorporated into Duvall's. The two references deal with ***vastly different*** issues (filtering subjectively objectionable e-mail versus preventing denial of service attacks) on ***different sides*** of the network (i.e., client versus server). Applicant respectfully submits that the Office's stated motivation to combine is hindsight reconstruction, which is an improper basis for a §103 rejection. Therefore, the Office has failed to establish a *prima facie* case of obviousness.

Furthermore, even if there were motivation to combine the two references (which there is not), the Office appears to mischaracterize the Fox reference. As noted above, Fox performs a ***literal string comparison*** between the domain name of a URL contained in a notification request and the domain name (or its immediate superdomain) listed in the accompanying site certificate. According to Fox, if there is not an ***exact match***, the notification request is denied. Applicant, on the other hand, claims a search pattern that can be used to detect an attack pattern. Applicant's search pattern is defined in a manner that permits ***variability*** among its constituent parts. Thus, the search pattern can include literal parts that call for an exact character-by-character match between those parts and corresponding parts of the input string (i.e., the type of literal string comparison that Fox discloses), and ***variable parts*** that allow for ***inexact matches or no match at all*** between those parts and corresponding parts of the input string. Fox does not disclose a search pattern that permits this type of variability. Accordingly, because even the ***improper*** combination of the Duvall and Fox references does not suggest the subject matter of this claim, this claim is allowable.

Claims 2-6 depend either directly or indirectly from claim 1 and are allowable as depending from an allowable base claim. These claims are also

allowable for their own recited features which, in combination with those recited in claim 1, are neither disclosed nor taught by the references of record, either singly or in combination with one another.

Claims 7-12

Claim 7 recites a Web server input string screening method comprising [emphasis added]:

- defining one or more search patterns that comprise literal characters and special characters, wherein the literal characters indicate exact characters in an input string that is intended for receipt by a Web server, and the special characters indicate ***variable characters*** in an input string that is intended for receipt by the Web server, the search patterns being usable to search for an attack pattern that can be used to attack the Web server, the attack pattern comprising content that is designed to constitute one or more of a disclosure attack, an integrity attack or a denial of service attack on the Web server; and
- storing the one or more search patterns in a memory location that is accessible to a screening tool for evaluating an input string that is intended for receipt by the Web server.

In making out the rejection of this claim, the Office again argues the combination of Duvall and Fox suggest this claim. Once more, Applicant respectfully submits that there is no motivation to combine the two references; and, in fact, Applicant is unclear how Fox's teachings could possibly be incorporated into Duvall's e-mail screening method. Therefore, the Office has failed to establish a *prima facie* case of obviousness.

In addition, Applicant respectfully submits that Fox does not disclose a search pattern that contains ***special (or variable)*** characters. Rather, as noted above, Fox simply utilizes literal string comparisons of the domain name specified in a URL and the domain name listed in an accompanying site certificate.

Accordingly, because even the ***improper*** combination of the Duvall and Fox references does not suggest the subject matter of this claim, this claim is allowable.

Claims 8-12 depend from claim 7 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which, in combination with those recited in claim 7, are neither disclosed nor taught by the references of record, either singly or in combination with one another.

In addition, with respect to claim 12, which is rejected in view of Oliver, that reference is not seen to add anything of significance given the allowability of this claim.

Claims 13-17

Claim 13 recites a Web server input string screening method comprising:

- defining one or more search patterns that are specified as a regular expression, the search patterns being usable to search for an attack pattern that can be used to attack the Web server, the attack pattern comprising content that is designed to constitute one or more of a disclosure attack, an integrity attack or a denial of service attack on the Web server; and
- storing the one or more search patterns in a memory location that is accessible to a screening tool for evaluating an input string that is intended for receipt by the Web server.

In making out the rejection of this claim, the Office again argues the combination of Duvall and Fox suggest this claim. Once more, Applicant respectfully submits that there is no motivation to combine the two references; and, in fact, Applicant is unclear how Fox's teachings could possibly be

incorporated into Duvall's e-mail screening method. Therefore, the Office has failed to establish a *prima facie* case of obviousness, and this claim is allowable.

Claims 14-17 depend from claim 13 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which, in combination with those recited in claim 13, are neither disclosed nor taught by the references of record, either singly or in combination with one another.

Claims 18-21

Claim 18 recites a Web server input string screening tool embodied on a computer-readable medium comprising [emphasis added]:

- a pattern matching engine that is configured to receive an input string that is intended for use by a Web server and evaluate the input string to ascertain whether it likely constitutes an attack on the Web server, the attack comprising one or more of a disclosure attack, an integrity attack or a denial of service attack on the Web server; and
- one or more patterns that are usable by the pattern matching engine to evaluate the input string, the patterns being defined in a manner that permits ***variability among the constituent parts*** of the one or more patterns.

In making out the rejection of this claim, the Office again argues the combination of Duvall and Fox suggest this claim. Once more, Applicant respectfully submits that there is no motivation to combine the two references; and, in fact, Applicant is unclear how Fox's teachings could possibly be incorporated into Duvall's e-mail screening method. Therefore, the Office has failed to establish a *prima facie* case of obviousness.

In addition, Applicant respectfully submits that Fox does not disclose a pattern that is defined in a manner that permits ***variability among its constituent parts***. Rather, as noted above, Fox simply utilizes literal string comparisons of the domain name specified in a URL and the domain name listed in an accompanying site certificate. Accordingly, because even the ***improper*** combination of the Duvall and Fox references does not suggest the subject matter of this claim, this claim is allowable.

Claims 19-21 depend from claim 18 either directly or indirectly and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which, in combination with those recited in claim 18, are neither disclosed nor taught by the references of record, either singly or in combination with one another.

Claims 22-25

Claim 22 recites one or more computer readable media having computer-readable instructions thereon which, when executed by a computer perform the following steps [emphasis added]:

- receiving an input string that is intended for use by a Web server;
- evaluating the input string using a search pattern to ascertain whether the input string contains an attack pattern that can be used to attack the Web server, the attack pattern comprising content that is designed to constitute one or more of a disclosure attack, an integrity attack or a denial of service attack on the Web server, the search pattern comprising literal characters and ***special characters***, wherein literal characters indicate exact characters in the input string, and the special characters indicate ***variable characters*** in the input string; and
- implementing a remedial action if an attack pattern is found that matches the search pattern.

In making out the rejection of this claim, the Office again argues the combination of Duvall and Fox suggest this claim. Once more, Applicant respectfully submits that there is no motivation to combine the two references; and, in fact, Applicant is unclear how Fox's teachings could possibly be incorporated into Duvall's e-mail screening method. Therefore, the Office has failed to establish a *prima facie* case of obviousness.

In addition, Applicant respectfully submits that Fox does not disclose a search pattern that contains ***special (or variable)*** characters. Rather, as noted above, Fox simply utilizes literal string comparisons of the domain name specified in a URL and the domain name listed in an accompanying site certificate. Accordingly, because even the ***improper*** combination of the Duvall and Fox references does not suggest the subject matter of this claim, this claim is allowable.

Claims 23-25 depend either directly or indirectly from claim 22 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which, in combination with those recited in claim 22, are neither disclosed nor taught by the references of record, either singly or in combination with one another.

Claims 26-31

Claim 26 recites a collection of Web server screening patterns embodied on a computer-readable medium comprising [emphasis added]:

- a memory; and
- a plurality of patterns stored in the memory, the patterns being useable to screen input strings that are intended for use by a Web

> server to ascertain whether the input strings comprise attack patterns, the attack patterns comprising content that is designed to constitute one or more of a disclosure attack, an integrity attack or a denial of service attack on the Web server, individual patterns being defined in a manner that permits ***variability among their constituent parts***.

In making out the rejection of this claim, the Office again argues the combination of Duvall and Fox suggest this claim. Once more, Applicant respectfully submits that there is no motivation to combine the two references; and, in fact, Applicant is unclear how Fox's teachings could possibly be incorporated into Duvall's e-mail screening method. Therefore, the Office has failed to establish a *prima facie* case of obviousness.

In addition, Applicant respectfully submits that Fox does not disclose a collection of Web server screening patterns where the individual patterns are defined in a manner that permits ***variability among their constituent parts***. Rather, as noted above, Fox simply utilizes literal string comparisons of the domain name specified in a URL and the domain name listed in an accompanying site certificate. Accordingly, because even the ***improper*** combination of the Duvall and Fox references does not suggest the subject matter of this claim, this claim is allowable.

Claims 27-31 depend from claim 26 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which, in combination with those recited in claim 26, are neither disclosed nor taught by the references of record, either singly or in combination with one another.

In addition, with respect to claim 31, which is rejected in view of Oliver, that reference is not seen to add anything of significance given the allowability of this claim.

New Claims

Claim 32 recites a Web server input string screening method comprising:

- determining an attack pattern that can be used to attack a Web server;
- defining a search pattern that can be used to detect the attack pattern, ***the search pattern being specified as a regular expression***;
- screening received input strings using the search pattern to ascertain whether the attack pattern is present; and
- implementing a remedial action if the search pattern is found to contain an attack pattern.

None of the references of record disclose or suggest the features of this claim. Accordingly, this claim is allowable.

Claim 33 depends from claim 32 and is allowable as depending from an allowable base claim. This claim is also allowable for its own recited features which, in combination with those recited in claim 32, are neither disclosed nor suggested by the references of record, either singly or in combination with one another.

Claim 34 recites one or more computer readable media having computer-readable instructions thereon which, when executed by a computer, perform the following steps:

- determining an attack pattern that can be used to attack a Web server;

- defining a search pattern that can be used to detect the attack pattern, ***the search pattern being specified as a regular expression***;
- screening received input strings using the search pattern to ascertain whether the attack pattern is present; and
- implementing a remedial action if the search pattern is found to contain an attack pattern.

None of the references of record disclose or suggest the features of this claim. Accordingly, this claim is allowable.

Conclusion

Applicant respectfully submits that all of the claims are in condition for allowance and Applicant respectfully requests a Notice of Allowability be issued forthwith. If the next anticipated action is to be anything other than issuance of a Notice of Allowability, Applicant respectfully requests a telephone call for the purpose of scheduling an interview.

Respectfully Submitted,

Dated: 8/4/04

By: _____

Lanee R. Sadler
Reg. No. 38,605
(509) 324-9256